

Japanese Patent Application Laid-open No. Hei 6-149563

(Partial Translation)

[0021]

Furthermore, an appropriate set of orders for an application can be selected if a structure is employed in which plural sets of orders are prepared in advance and each number for each set of orders is corresponded to one of data for decoding so that by selecting the data an appropriate set is selectable among the plural sets of orders.

[0029]

..An instruction register (hereinafter referred to as "IREG") 6 stores an output data of the IMEM 4 with its address being specified by the address register 2, which is an order to be followed (consisting of operand only) or an operation code modifying order. An instruction decoder (hereinafter referred to as "IDEC") 8a inputs therein an order to be decoded from the IREG 6, and decodes the order by a specified decoding method. Here, the IDEC 8a does not have an fixed order decoding method in itself, so that it inputs therein an operation code 22 which is the data for decoding stored in the storage device 20, and decides a decoding method based on that data. For example, a structure can be taken in which an exchanging device is used that can switch the allocation correspondence between a specified position of a bit constituting the order (i.e., operand) and a hardware control line into an specified pattern based on the data for decoding. Then, the inputted

order is decoded by the selected decoding method, and a control signal 12 is outputted to a controlled section 10 which is related to that particular order. For an instance, when the data provided from the storage device 20 shows that the operand is an order of the C system, respective control signals are outputted to each controlled section for the three controls of the SQM, AAU, and DT2.

[0030]

Here, when it is detected from a flag of the order or so forth, that the order inputted to the IDEC 8 is an operating code modification order, the IDEC 8a outputs a control signal 14 to a control line for a writing device 23, whereby the writing device 23 gets operated and rewrites the data stored in the storage device 20, i.e., the operation code. For example, when the operating code modification order is the operating code modification order X561a, the data stored in the storage device 20, i.e., the operation code, is modified to a data which corresponds to the order of the B system. Thereafter, an order inputted to the IDEC8a is recognized as an order of B system and decoded.

(19)日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11)特許出願公開番号

特開平6-149563

(43)公開日 平成 6 年(1994) 5 月27日

(51)Int.Cl.<sup>5</sup>

G 0 6 F 9/30

識別記号

3 1 0 E 9189-5B

庁内整理番号

F I

技術表示箇所

審査請求 未請求 請求項の数14(全 14 頁)

(21)出願番号 特願平4-296002  
(22)出願日 平成 4 年(1992)11月 5 日

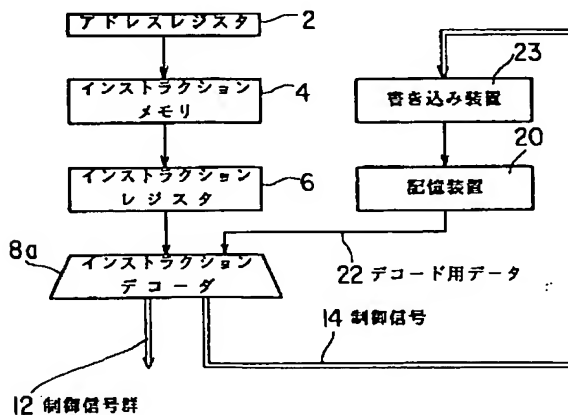
(71)出願人 000003078  
株式会社東芝  
神奈川県川崎市幸区堀川町72番地  
(72)発明者 児山 元昭  
神奈川県川崎市幸区堀川町580番 1 号 株  
式会社東芝半導体システム技術センター内  
(74)代理人 弁理士 鈴江 武彦

(54)【発明の名称】 データ処理装置

(57)【要約】

【目的】従来のプログラム内蔵型のデータ処理装置より処理能力が高く、開発期間が短く、フレキシビリティに富み、汎用性を有し、開発コストを下げたデータ処理装置を提供することを目的とする。

【構成】インストラクションデコーダにおけるデコード方法を決定するための少なくとも1つのデコード用データを保持する記憶回路を設けることによって、前記インストラクションデコーダはインストラクションレジスタから入力される命令をデコードする際に、前記記憶回路から前記データを取り込み、その少なくとも1つのデコード用データに基づいてまたはその少なくとも1つのデコード用データと前記入力した命令に含まれるオペコードとに基づいてデコード方法を決定する。前記記憶装置の内容は、デコード用データの変更命令によって書き替えるか、本システムの外部から書き替えるか、あるいは予め固定される。



【特許請求の範囲】

【請求項1】命令列を格納するための第1の記憶手段と、  
次に実行すべき命令が格納されている前記第1の記憶手段のアドレスを示すためのアドレス指示手段と、  
前記アドレス指示手段により前記第1の記憶手段から与えられる前記命令を保持するための第2の記憶手段と、  
前記第2の記憶手段に保持されている命令の復号方法に関連する復号用コードを保持するための第3の記憶手段と、  
命令の復号方法が固定されていない復号手段であって、  
前記第3の記憶手段から復号用コードを入力し、その復号用コードに基づいて復号方法を決定し、前記第2の記憶手段から命令を入力し、前記命令を復号し、前記命令に関連する制御信号を出力するための復号手段とを有することを特徴とするデータ処理装置。

【請求項2】命令列を格納するための第1の記憶手段と、  
次に実行すべき命令が格納されている前記第1の記憶手段のアドレスを示すためのアドレス指示手段と、  
前記アドレス指示手段により前記第1の記憶手段から与えられる前記命令を保持するための第2の記憶手段と、  
前記第2の記憶手段に保持されている命令の復号方法に関連する復号用コードを保持するための第3の記憶手段と、  
命令の復号方法が固定されていない復号手段であって、  
前記第3の記憶手段から復号用コードを入力しおよび前記第2の記憶手段から命令を入力し、その復号用コードと命令自体とに基づいて復号方法を決定し、前記命令を復号し、前記命令に関連する制御信号を出力するための復号手段とを有することを特徴とするデータ処理装置。

【請求項3】前記命令列は前記復号用コードを書き替えるための書替命令を含み、前記装置は、前記書替命令に  
応答して前記復号用コードを書き替えるための書替手段をさらに有することを特徴とする請求項1または2に記載の装置。

【請求項4】前記命令列の各命令はオペランドのみよりなり、前記復号用コードは前記オペランドのみよりなる命令に関連する命令コードであり、  
前記命令列は前記復号用コードを書き替えるための書替命令を含み、前記装置は、前記書替命令に  
応答して前記命令コードを書き替えるための書替手段をさらに有することを特徴とする請求項1に記載の装置。

【請求項5】前記命令列の各命令はそれぞれオペランドのみよりなる命令または命令コードおよびオペランドよりなる命令のいずれかよりなり、  
前記命令列は前記復号用コードを書き替えるための書替命令を含み、前記装置は、前記書替命令に  
応答して前記復号用コードを書き替えるための書替手段をさらに有し、

前記復号用コードは、関連する命令がオペランドのみよりなる命令であるかまたはオペランドおよび命令コードよりなる命令であることを示す情報を含み、前記復号手段は、前記命令がオペランドのみよりなる命令である場合に前記復号用コードのみに基づいて復号方法を決定し、  
前記命令がオペランドおよび命令コードよりなる命令である場合に前記復号用コードと前記命令コードとに基づいて復号方法を決定することを特徴とする請求項2に記載のデータ処理装置。

10 【請求項6】前記アドレス指示手段が保持するアドレスデータのうちの所定部分のデータに  
応答して前記命令コードを逐次あるいは変更があるときのみ書き替えるための書替手段をさらに有することを特徴とする請求項1または2に記載の装置。

【請求項7】前記復号用コードは前記装置の外部から設定されることを特徴とする請求項1または2に記載の装置。

20 【請求項8】前記第3の記憶手段は読み出し専用の記憶手段であって、前記復号用コードは  
予め選択されたデータに固定されることを特徴とする請求項2に記載の装置。

【請求項9】前記第3の記憶手段は再書き込みが可能な読み出し専用の記憶手段であって、前記復号用コードは  
予め選択されたデータに固定され、さらに随時前記固定された復号用コードは変更および再固定されることを特徴とする請求項2に記載の装置。

30 【請求項10】前記復号手段は前記第3の記憶手段から前記復号用データを入力する代わりに、前記アドレスデータのうちの所定部分のデータを復号用データとして入力し、前記復号方法を決定することを特徴とする請求項1または2に記載の装置。

【請求項11】命令列を格納するための第1の記憶手段と、  
次に実行すべき命令が格納されている前記第1の記憶手段のアドレスを示すためのアドレス指示手段と、  
前記アドレス指示手段により前記第1の記憶手段から与えられる前記命令を保持するための第2の記憶手段と、  
前記第2の記憶手段に保持されている命令の復号方法に関連する第1および第2の復号用コードを保持するための第3の記憶手段と、  
命令の復号方法が固定されていない復号手段であって、  
前記第3の記憶手段から第1および第2の復号用コードを入力し、それらの復号用コードに基づいて復号方法を決定し、  
前記第2の記憶手段から命令を入力し、前記命令を復号し、前記命令に関連する制御信号を出力するための復号手段とを有することを特徴とするデータ処理装置。

40 【請求項12】前記命令列は、前記第1の復号用コードを書き替えるための書替命令を含み、前記装置は、前記書替命令に  
応答して前記復号用コードを書き替えるための書替手段をさらに有し、

50 【請求項13】前記命令列は、前記第1の復号用コードを書き替えるための書替命令を含み、前記装置は、前記書替命令に  
応答して前記復号用コードを書き替えるための書替手段をさらに有し、

の書替手段をさらに有することを特徴とする請求項11に記載の装置。

【請求項13】前記第2の復号用コードは予め選択されたデータに固定されることを特徴とする請求項11または12に記載の装置。

【請求項14】前記第1の記憶手段はインストラクションメモリまたはプログラムカウンタであり、前記第2の記憶手段はインストラクションレジスタであり、前記アドレス指示手段はアドレスレジスタであり、前記復号手段はインストラクションデコーダであることを特徴とする請求項1ないし13のいずれか1項に記載の装置。

【発明の詳細な説明】

【0001】

【産業上の利用分野】本発明は、記憶装置に貯えられた命令に従って動作するデータ処理装置に関する。

【0002】

【従来の技術】一般に、データ処理装置は布線論理で処理を行う布線論理型装置と、予め処理内容すなわちプログラムを記憶装置に貯えておき、そのプログラムに従って処理を行うプログラム内蔵型装置の2つに大きく分けられる。

【0003】布線論理型のデータ処理装置の中で集積回路に関しては、製品毎に全て作り直すフルカスタムICと、IC製造工程の一部を標準品として用意することで開発期間を前者より短くしたセミカスタムICがある。これらのICは所望の処理に適した回路構成で並列処理

を行えるため高速処理性能が得られる。さらに、ある処理を一定時間で実行すればよいときは、クロックを遅くして消費電力を小さくできる。しかし、その反面、処理内容を固定しているために異なった処理を実行させることはできず、その際は新たな開発つまり配線、タイミング設計のし直しおよびICの製造が必要となる。

【0004】一方、プログラム内蔵型のデータ処理装置の中で集積回路に関しては、MCU (Micro Controller Unit) や近年デジタル信号処理の分野などで頻繁に用いられてきているDSP (Digital Signal Processor) などがある。MCUまたはDSPなどは、プログラムを書き替えるだけで所望の処理を実行できるのが大きな利点である。また、汎用性を有する量産品であるために低コストである。しかしそのように汎用性がある反面、処理において無駄な動作が発生しており、その分処理速度が低下するのは避けられない。ここで、前記各種集積回路の特質を前記各項目について表1に示す。矢印の方向が望ましい方向を示す。

【0005】表1のように各特質の間にはトレードオフの関係が存在し、従って、アプリケーションの開発において前記各種集積回路のうちのいずれを採用するかを決定するときに、それぞれの開発期間、フレキシビリティ、演算処理能力、コストについて総合的に評価することが重要である。

【0006】

【表1】

	フルカスタムIC	セミカスタムIC	DSP	MCU	CPU
アプリケーション開発期間	長	短	↑	↑	↑
汎用性・フレキシビリティ	低	高	↑	↑	↑
アプリケーション処理能力/同一クロック	高	低	↓	↑	↑
少量アプリケーション時のコスト	高	高	↑	↑	↑

【0007】しかし、その際あまり選択の余地がないことも少なくない。例えば、単位時間内に大量の演算が必要な場合はカスタムICを採用せざるを得ない。従って、この場合には高いコストが不可避となる。また、フレキシビリティあるいは低コストを必要とする場合はMPUなどを採用することになり、演算処理能力がある程度犠牲とならざるを得ない。すなわち、短期間かつ低コストで、フレキシビリティに富み、複雑で大量の演算量を必要としたアプリケーションを開発するのは困難であった。従って、前記評価点のいずれの観点からも十分に要求を満たすICを開発、提供することは非常に重要な

ことである。

【0008】表1のいずれの項目においても中程度に位置するDSPは、そのような視点から、用途あるいは分野をある程度特定化した上で、フレキシビリティと比較的高い処理能力との両方を同時に確保することの特徴としたものである。

【0009】従来のDSPの中にはマイクロプログラム方式を取るものがある。ユーザインターフェースとしてのプログラムコードは単一でもそれを複数マシンサイクルとして展開するマイクロプログラムROMを用意することで、ユーザプログラムコードの互換性を保ち、かつ

複合命令を単一コードで表わせることでユーザによるプログラムの保守性を保てるという利点や、命令体系を変更する場合、マイクロコードを変更するだけで良いという利点がある。しかし、この方式のものではマシンサイクル数が増加するために高速処理は困難であった。

【0010】また、水平プログラム型と呼ばれるタイプのDSPがある。この型のものは、内部にデータバスとプログラムバスを有し、メモリの内容の積和演算を並列処理して、高速処理が行えるようになっている。しかし、命令セットは有限の命令群に分けられ、ハードウェアの制御線をデコードし易いように各命令にはプログラムメモリデータビット幅の中の所定のビットフィールドが固定的に割り付けられており、所望の並列動作機能が当該DSPに含まれない場合は実際に存在する適当な命令を逐次処理で実行して対処せざるを得ないので途端に性能の低下が生じてしまう。さらに、所定の性能の確保が必要な場合に、望まれる並列動作機能を含むように当該DSPの仕様を若干変更して所望のDSPの開発を行おうとしても、前記固定的に割り付けられたビットフィールドの再設計の必要性のためにその開発は容易ではな

かった。

【0011】また、単一マシンサイクルでの全ハードウェアの並列動作を可能にして性能を向上させることが考えられるが、実際には常に全ハードウェア性能を並列動作させることは少ないため、全命令ビット中で有効に使われているビットは少なく、ハードウェアの無駄が大きい。従って、このことを考慮し、ハードウェア的に並列可能な全機能を一命令中に納めることをやめ、頻繁に用いられる機能を適宜組み合わせで有限語長に納めることが多い。この場合は積和演算以外の処理にとっては必ずしも最適化がなされた構造ではなく、十分に性能が引き出せていたとは言えない。

【0012】このように、DSPにおいても、開発期間、フレキシビリティ、演算処理能力、コストのどの点からとも要求を満たすICを十分に提供しているとは言えないのが現状である。

【0013】

【発明が解決しようとする課題】この発明は、上記問題を考慮してなされたもので、従来のプログラム内蔵型のデータ処理装置より処理能力が高く、開発期間が短く、フレキシビリティに富み、汎用性を有し、開発コストを下げたデータ処理装置を提供することを目的とする。

【0014】

【課題を解決するための手段】本発明は、DSPにおいては同一のオペレーションコードを有する命令が連続して実行される傾向にあることに着目してなされたものである。

【0015】本発明における課題解決のための第1の手段は、命令列を格納するための第1の記憶手段と、次に実行すべき命令が格納されている前記第1の記憶手段の

アドレスを示すためのアドレス指示手段と、前記アドレス指示手段により前記第1の記憶手段から与えられる前記命令を保持するための第2の記憶手段と、前記第2の記憶手段に保持されている命令の復号方法に関連する復号用コードを保持するための第3の記憶手段と、命令の復号方法が固定されていない復号手段であって、前記第3の記憶手段から復号用コードを入力し、その復号用コードに基づいて復号方法を決定し、前記第2の記憶手段から命令を入力し、前記命令を復号し、前記命令に関連する制御信号を出力するための復号手段とを有することを特徴とする。

【0016】本発明における課題解決のための第2の手段は、命令列を格納するための第1の記憶手段と、次に実行すべき命令が格納されている前記第1の記憶手段のアドレスを示すためのアドレス指示手段と、前記アドレス指示手段により前記第1の記憶手段から与えられる前記命令を保持するための第2の記憶手段と、前記第2の記憶手段に保持されている命令の復号方法に関連する復号用コードを保持するための第3の記憶手段と、命令の復号方法が固定されていない復号手段であって、前記第3の記憶手段から復号用コードを入力しおよび前記第2の記憶手段から命令を入力し、その復号用コードと命令自体とに基づいて復号方法を決定し、前記命令を復号し、前記命令に関連する制御信号を出力するための復号手段とを有することを特徴とする。

【0017】本発明における課題解決のための第3の手段は、命令列を格納するための第1の記憶手段と、次に実行すべき命令が格納されている前記第1の記憶手段のアドレスを示すためのアドレス指示手段と、前記アドレス指示手段により前記第1の記憶手段から与えられる前記命令を保持するための第2の記憶手段と、前記第2の記憶手段に保持されている命令の復号方法に関連する第1および第2の復号用コードを保持するための第3の記憶手段と、命令の復号方法が固定されていない復号手段であって、前記第3の記憶手段から第1および第2の復号用コードを入力し、それらの復号用コードに基づいて復号方法を決定し、前記第2の記憶手段から命令を入力し、前記命令を復号し、前記命令に関連する制御信号を出力するための復号手段とを有することを特徴とするデータ処理装置。

【0018】

【作用】本発明においては、インストラクションデコーダにおけるデコード方法を決定するための少なくとも1つのデコード用データを保持する記憶回路を設けることによって、前記インストラクションデコーダはインストラクションレジスタから入力される命令をデコードする際に、前記記憶回路から少なくとも1つのデコード用データを取り込み、その少なくとも1つのデコード用データに基づいてまたはその少なくとも1つのデコード用データと前記入力した命令に含まれるオペコードとに基づ

いてデコード方法を変更する。

【0019】前記記憶装置の内容を変えることで、各ビットフィールドとハードウェアの制御線を自由に組み替えることにより、単一命令中に並列制御できる機能ブロック種の組み合わせを自由に変えることができる。従って、従来よりも大きな汎用性、フレキシビリティ、高速性、さらに設計変更の容易性を与える。前記記憶装置の内容は、デコード用データの変更命令によって書き替えるか、本システムの外部から書き替えるか、あるいは予め固定される。

【0020】命令をオペランドのみにより構成し、オペコードを前記記憶装置に保持させ、オペコードが変わるときのみオペコード変更命令を実行させるように構成すると、従来のオペコードとオペランドとからなる命令を用いた構成に比べて、メモリの容量を小さくでき、コストダウンおよびチップサイズの小型化による高速化が可能となる。

【0021】また、予め複数の命令セットを用意しておき、その命令セットの番号に前記デコード用データのうちの1つのデータに対応させ、そのデータを設定することにより、前記複数の命令セットのうちいずれかの命令セットを用いるかを適宜設定するように構成すれば、アプリケーションにより適宜命令セットを選択できる。

【0022】

【実施例】以下本発明における実施例を図面を参照しながら説明する。なお、説明の簡潔化のためにそれぞれ同一部分については同一符号を付して説明を省略する。また、以下の説明においては、命令という語句はいわゆる実行の単位であるインストラクションを示し、前記命令中に含まれ並列処理される個々の命令のことは制御という。

【0023】図1は本発明に係るデータ処理装置の第1の実施例の構成を示すブロック図である。インストラクションメモリ（以下IMEMという）4には、命令の列からなるプログラムが記憶されている。

【0024】まず、このIMEM4に格納される命令の構成を説明する。従来の命令は図2の命令70のように命令の種類を判別するためのオペコードとそのオペコードで示された操作を施す対象を指定するオペランドとから構成されるが、DSPでのプログラムの実行においては同じオペコードの命令が連続して実行される傾向にあり、その連続する同一オペコードの情報としての冗長性に着目し、本発明の第1の実施例では、命令を図2の命令72のようにオペランドのみより構成し、分離されたオペコードは、図1の記憶装置20に格納し、次に実行すべき命令のオペコードが変わるときのみ、その情報を与え、前記記憶装置20のデータを更新する構成を導入した。

【0025】図3に本実施例における3つ命令例を示す。図3の命令中のオペランドは各被制御部に対する制

御を指定する。A体系の命令74は即値IDをメモリに転送する制御(DST)であり、B体系の命令76はALU(Arithmetic Logic

【0026】Unit)動作制御、AAU(データメモリアドレッシング)動作制御、DPR(Data Pipeline Register)0、DPR1へのデータ転送制御DT1、ACCからDT2フィールドで指定されるレジスタメモリへのデータ転送DT2を同時に指定する命令であり、C体系の命令78はシーケンス制御(サブルーチンコール等)SQM、前記AAU、汎用バス転送DT2を同時に指定する命令である。

【0027】なお、この他のフィールドの組み合わせも可能である。また、上記A、B、Cの3通りより多い命令体系でも少ない命令体系でも差し支えない。さらに、シーケンス制御(リターン等)RTM、パラレル入出力制御PIO(Parallel Inputand Output)などの他の制御を適宜組み込んでもよいし、命令ビットに空きがあっても差し支えない。また、命令長はまったく任意である。

20 【0028】さらに、オペコードを分離し記憶装置20に格納する構造を用いるために、前記命令セットに加えて以下のようなオペコード変更命令を導入した。オペコード変更命令は前記記憶装置20にセットされているデータすなわちオペコードをある体系に対応するデータから他の体系に対応するデータに変更するための命令である。例えば、図4に示すインストラクションメモリマップ24aのように、A体系の命令60aがいくつか連続して実行され、次にB体系の命令60bがいくつか連続して実行され、最後にC体系の命令60cがいくつか連続して実行されるとする。A体系の命令に対応するオペコードは最初のA体系の命令の実行直前に前記記憶装置20にセットされ以後保持される。オペコード変更命令X56aは、前記記憶装置20にセットされているデータすなわちオペコードをB体系の命令に対応するデータに書き替えるために、前記連続するA体系の最後の命令と連続するB体系の最初の命令との間に挿入される。オペコード変更命令Y56bも同様に、前記記憶装置20にセットされているデータすなわちオペコードをC体系の命令に対応するデータに書き替える。なお、プログラムにおいて一番始めに実行される命令に対するオペコードの設定に関しては、前記記憶装置20の初期値を固定してもよいし、初期値設定用の命令をさらに導入してもよいし、また前記オペコード変更命令を共用してもよい。

【0029】次に、図1の構成および動作について説明する。アドレスレジスタ2は、前記IMEM4に記憶されているプログラムに関し、次に実行すべき命令が格納されているアドレスを指し示す。インストラクションレジスタ(以下IREGという)6は、アドレスレジスタ2によってアドレスを指定されるIMEM4の出力デー



タつまり次に実行すべき（前述のようなオペランドのみの）命令あるいは前記オペコード変更命令のいずれかを貯える。インストラクションデコーダ（以下IDECという）8aは、前記IREG6よりデコードすべき前記命令を入力し、所定のデコード方法で前記命令をデコードする。ここで、前記IDEC8a自体には、命令のデコード方法が固定されておらず、前述した記憶装置20に保持されているデコード用データつまりオペコード22を入力し、そのデータに基づいてデコード方法を決定する。例えば、命令を構成するビットの所定の位置（つまりオペランド）と、ハードウェアの制御線との割り付けの対応を前記デコード用データにより所定のパターンに切替えらる交換装置を用いて行う構成を取ってもよい。そして、前記決定したデコード方法で前記入力した命令をデコードし、その命令に関連する被制御部10への制御信号12を出力する。例えば、記憶装置20から与えられたデータが、当該オペランドはC体系の命令であることを示しているときは、前記SQM、AAU、DT2の3つの制御に対する各被制御部にそれぞれ制御信号を出力する。

【0030】ここで、前記IDEC8aが前記入力された命令はオペコード変更命令であることを前記命令のフラグなどにより検出した場合、前記IDEC8aは書き込み装置23に対する制御線に制御信号14を出力し、前記書き込み装置23を動作させ、前記記憶装置20に記憶されているデータつまりオペコードを書き替えさせる。例えば、前記オペコード変更命令が図4のオペコード変更命令X56aであるとき、前記記憶装置20に記憶されているデータつまりオペコードをB体系の命令に対応するデータに変更する。以後、IDEC8aに入力される命令はB体系の命令であると認識されてデコードされる。

【0031】このように構成すると、従来の図2の命令70を用いた構成に比べて、同じ命令体系で図3の命令72のような構成をとった場合、IMEMのビット幅が従来のオペコードのビット幅分だけ少なくてすむ。以上のような実施例のデータ処理装置では、IMEMのビット幅を小さくしたためメモリ総ビット数を減らすことができる。従来の命令においてオペコードのビットの占める割合が高いほど、また、IMEM4に格納する命令数が多いほど、メモリ削減の効果は大きい。まず、メモリの容量を小さくできると、コストダウンが可能となる。次に、それによってチップサイズが小さくなるので、クロックを早くすることができ、処理の高速化が可能となる。また、メモリの容量が同じである場合には、従来のオペコードに割り当てられていたビットフィールドに新たに他のオペランドを追加できるので、従来よりも並列実行できる命令の数が増し、設計の自由度の向上および高機能を与える。

【0032】また、従来のようにハードウェアの制御線

をデコードし易いように各命令のデータビットの中の所定のビットフィールドに固定的に割り付けるのではなく、前記記憶装置20の内容を変えることで、各ビットフィールドとハードウェアの制御線を自由に組み替えることができるので、単一命令中に並列制御できる機能ブロック種の組み合わせを自由に定めることができ、従来よりも大きなフレキシビリティを与える。

【0033】また、従来は、並列処理できる命令のすべての組み合わせを実行できる命令を用意することは難しく、従って、並列処理させたいすべての命令を含む命令が用意されていないことも多く、例えば、AとCの機能を実現させたいとき、AとBを並列処理できる命令と、AとCを並列処理できる命令との2つしか予め用意されていない場合に前記2つの命令の順次処理が必要であったことと比較すると、同じ命令ビット長で並列処理可能で且つ並列処理させたい機能を同一命令命令中で行えるので高速処理が行える。特にループ処理中で同一命令を繰り返し用いる場合、回数が大きいほど効果が大きい。

【0034】さらに、システムを開発する際に、効率的な命令ビット構成を持つ命令を容易に選択／追加することができ、加えてあるアプリケーションでは並列処理を行わせる必要のない機能の組み合わせを持つ命令構成を除くことで、無駄な命令セットを除去でき、また、必要な命令種が少ないアプリケーション用にはIMEMの容量を小さくすることが可能である。当然、システムのマイナーチェンジも容易である。次に、本発明における第2の実施例について説明する。図5は本発明の第2の実施例の構成を示すブロック図である。

【0035】構成図は図1と同様であり、IDEC8b以外のブロックの動作も図1と同様であるので、同一部分については同一符号を付して説明を簡略化し、IDEC8bに関する動作および命令構成のみ説明する。

【0036】図6は本発明の第2の実施例における命令のビット構成例である。ここでは、記憶装置20に保持されるデータをOPREGとする。図6で示される命令構成ではOPREGの値によって命令系が3つに分かれる。もちろん、2つでも、3つ以上でも差し支えない。OPREG=1および2のときの命令82aおよび命令82bは図3と同様にオペコードのないオペランドのみの構成であり、前記OPREGの値がオペコードを与える。一方、OPREG=0のときはオペコード1～オペコード4の4種類のオペコードを持つ命令80a、命令80b、命令80cおよび命令80dからなる従来と同様の命令体系となる。つまり、前記OPREGの値には、命令がオペランドのみからなるか、オペコードおよびオペランドからなるかを示す情報が含まれており、OPREG=1および2のときは、さらにデコード方法を示す情報が含まれている。

【0037】このような命令を用いて、例えば、図7に示すインストラクションメモリマップ24bのようなブ

プログラムが組まれる。ここで、D体系の命令62aが連続して実行され、次にF体系の命令62bが連続して実行され、最後にE体系の命令62cが連続して実行されるとする。D体系の命令に対応するオペコードは最初のD体系の命令の実行直前に前記記憶装置20にセットされ、以後D体系の命令が連続する間保持される。オペコード変更命令U58aは、前記記憶装置20にセットされているデータすなわちオペコードをF体系の命令に対応するデータに書き替えるために、前記連続するD体系の最後の命令と連続するF体系の最初の命令との間に挿入される。オペコード変更命令V58bも同様に、前記記憶装置20にセットされているデータすなわちオペコードをE体系の命令に対応するデータに書き替える。

【0038】つぎに、IDEC8bを以下のように動作させる。IDEC8bは、IREG6よりデコードすべき命令を入力し、所定のデコード方法で前記命令をデコードする。ここで、前記IDEC8aは、前述した記憶装置20に保持されているデコード用データ22を入力し、そのデータすなわちOPREGの値に基づいてデコード方法を決定する。OPREG=1の場合は、そのOPREGの値“1”に対応するデコード方法つまりD体系の命令としてデコードする。オペコード変更命令UによりOPREG=0となった場合は、以後オペコードが変更されるまで前記入力した命令はオペコードを有することを示しており、その命令に含まれるオペコード例えば命令80aではオペコード1によりデコードする。オペコード変更命令UによりOPREG=2となった場合は、同様にE体系の命令としてデコードする。そして、関連する制御信号を出力する。

【0039】上記構成をとると次のような利点がある。すべてオペランドのみの命令で構成した場合に同一オペコードが連続せずに入れ替わるときはかえってオペコード変更命令の実行分むだが生ずることになるので、そのような入れ替わることが避けられない場合は、連続実行する確率の高い命令には、命令82aあるいは命令82bのようにオペランドのみよりなる命令を割り当て、オペコード分の余分ビットのない構成を取り、また、連続実行する確率が低く頻繁にオペコードが入れ替わる確率の高い命令には、命令80a~80dのようにオペコードおよびオペランドからなる命令を割り当てることにより、それ自身が持つオペコードによってデコードする。これによって、むだなオペコード変更命令の発生を回避することができ、さらなる高速化ができる。

【0040】もちろん、OPREGの割り当てを0と1の他に00と11等、異なる符号化方法で符号化してもよい。また、オペコードの数も上記は4種類であるが、それ以上でもそれ以下でも差し支えない。オペコード1として1ビット0を、オペコード2に対して3ビット110を、オペコード3に対して2ビット10を、オペコード4に対して3ビット111のようにハフマン符号な

どの可変長のコードを付与してもよいし、同じ4種類に対して00、01、10、11の様に等しく2ビットのバイナリコードを与えてもよい。また、並列同時制御する組み合わせ、命令のビット長も適宜選択可能である。

【0041】図8は、本発明の第3の実施例の構成を示すブロック図である。アドレスレジスタ2、IMEM4、IREG6については図1および5と同様であり、同一符号を付して説明を簡略化する。

【0042】本実施例では、IDEC8cのデコード方法を決定するデータ22をプログラムでは変更できなくし、記憶装置30に固定する。データの固定方法15としては、ROMによる書き込みやジャンパ線、ディップスイッチによる固定、半導体集積回路のアルミ配線、PLD等いずれの方法でも可能である。

【0043】まず、予め図9に示すような、図6と同様にオペコードを有する4つの命令群80a~dおよび全並列可能機能ブロック制御フィールドをそなえた命令84を構成しておく。

【0044】つぎに、IDEC8cを以下のように動作させる。IDEC8cは、IREG6よりデコードすべき命令を入力し、所定のデコード方法で前記命令をデコードする。ここで、前記IDEC8cは、前述した記憶装置30に保持されているデコード用データ22を入力し、そのデータすなわちOPREGの値に基づいてデコード方法を決定する。OPREG=0である場合は、前記入力した命令はオペコードを有することを示しており、そのオペコード例えば命令80aではオペコード1によりデコードする。OPREG=1の場合は、そのOPREGの値“1”に対応するデコード方法でデコードする。そして、関連する制御信号を出力する。

【0045】上記構成を取ると、使用する際にOPREGの値を固定することによって、従来と同様のすべての命令がオペコードを有するシステムあるいは全並列可能機能ブロック制御フィールドをそなえた命令のシステムのいずれかとして使用することができる。

【0046】前記記憶装置20のデータつまりOPREGは、使用者の都合に応じて適宜設定、変更できるようにしてもよい。例えば、IMEMアドレス空間が許容量の範囲内に収まる場合は命令84を用いることができ、データ処理能力は大きくなるので有効である。また、メーカーの方が出荷前に設定してもよい。この場合、1種のICを複数種類の製品として販売することができるので、コストなどの面から有利である。

【0047】また、データ処理装置の中でIMEMの語長と、記憶装置の内容を変えるだけで、所望の並列可能フィールドの組み合わせと、命令体系の組み合わせを作れるため、異なったアプリケーションに適した同時並列処理可能フィールドの組み合わせのデータ処理装置を短い開発期間で製造できるという利点がある。

【0048】上記記憶装置20の実現方法としては、R

OM (Read Only Memory)、E<sup>2</sup> PROM (Electrical Erasable Programmable ROM)、PLD (Programmable Logic Device)、半導体集積回路におけるアルミ配線によるデータ記述等で実現可能である。

【0049】本発明における第4の実施例は、IDECと記憶装置以外の部分は第1または第2の実施例と同一構成、同一動作であり説明を省略する。本実施例においては前記IDECのデコード用データを保持する記憶装置に第1のデータおよび第2のデータを持たせる。ここでは、前記IDECは前記記憶装置の第1のデータおよび第2のデータを基にデコード方法を決定する。そして、そのデコード方法によってIREGより入力した命令をデコードし、関連する制御信号を出力する。前記第1のデータは前述したいずれかの実施例のようにデコード方法を決定するためのデータとして用いる。さらに、それに対応する命令セットを複数用意し、前記第2のデータが複数用意された命令セットのうちどの命令セットを用いるかを指定するように構成してもよい。前記記憶装置の第2のデータは、アプリケーションに応じて使用者がダイナミックに用いてもよいし、適宜設定、変更できるようにしてもよい。また、メーカーの方が出荷前に設定してもよい。この場合、1種のICを複数種類の製品として販売することができるので、コストなどの面から有利である。

【0050】また、データ処理装置の中でIMEMの語長と、記憶装置の内容を変えるだけで、所望の並列可能フィールドの組み合わせと、命令体系の組み合わせを作るため、異なったアプリケーションに適した同時並列処理可能フィールドの組み合わせのデータ処理装置を短い開発期間で製造できるという利点がある。

【0051】上記記憶装置の第2のデータを予め固定する場合は、ROM (Read Only Memory)、E<sup>2</sup> PROM (Electrical Erasable Programmable ROM)、PLD (Programmable Logic Device)、半導体集積回路におけるアルミ配線によるデータ記述等で実現可能である。図10は、本発明の第5の実施例の構成を示すブロック図である。図1と同一部分については同一符号を付し説明を省略する。

【0052】この装置においては、IMEM4のインストラクションメモリマップは、例えば図12のインストラクションメモリマップ24cのように構成する。つまり、アドレスレジスタ2のビット構成68aの上位2ビットによって命令の体系が識別されるように、ジャンプ命令を用いて各命令が配置される。図12では前記アドレスの上位ビットが00のエリアがG体系の命令64aに対応し、上位ビットが01、10、11のエリアがそれぞれH、I、J体系の命令64a、b、cに対応する。

【0053】記憶装置21はアドレスレジスタの上位ビットをアドレスとするメモリ空間を有しており、前記ア

ドレスレジスタの上位ビットにより示された記憶装置21のアドレスに記憶されているデコード用データがIDEC8dに与えられる。前記IDEC8dはそのデコード用データによってデコード方法を決定し、IREG6より入力した命令をデコードし、関連する制御信号を出力する。

【0054】このような構成とすると前記4種の命令体系をOPCODE変換命令を使う代わりに、ページ間のジャンプ命令で代用できるという利点がある。なお、記憶装置へのビット割り当てを命令系により可変長にして、各命令系へのメモリマップ割り当てを均等ではなく適宜容量を設定してもよい。

【0055】図11は、本発明の第6の実施例の構成を示すブロック図である。図10と同一部分については同一符号を付し説明を省略する。この装置においても、IMEM4のインストラクションメモリマップは図12のインストラクションメモリマップ24cのような構成にする。つまり、IMEM4のアドレス68bの上位2ビットによって命令の体系が識別されるように、ジャンプ命令を用いて各命令が配置される。図12では前記アドレスの上位ビットが00のエリアがG体系の命令64aに対応し、上位ビットが01、10、11のエリアがそれぞれH、I、J体系の命令64a、b、cに対応する。

【0056】ここで、アドレスレジスタはIDEC8eのためのデコード用データの記憶装置としての部分を兼ねた構成となっている。そして、アドレスレジスタの上位ビットのデータが直接IDEC8dに与えられる。前記IDEC8dはそのデコード用データによってデコード方法を決定し、IREG6より入力した命令をデコードし、関連する制御信号を出力する。このような構成にしても前記4種の命令体系をオペコード変換命令を使う代わりに、ページ間のジャンプ命令で代用できるという利点がある。なお、記憶装置へのビット割り当てを命令系により可変長にして、各命令系へのメモリマップ割り当てを均等ではなく適宜容量を設定してもよい。

【0057】図13に、本発明に係るインストラクションデコーダ（以下デコード回路という）の第1の実施例の構成を示す。デコード回路16は、復号すべき命令を構成するオペランドを入力する端子と、各被制御部に対する制御に関連する回路に連結する出力端子と、記憶装置20に保持されている復号用データおよび復号すべき命令がオペコードを含むときはそのオペコードも用いて決定されたパターンに従って各入力端子と各出力端子とを連結する交換装置とを用いたビットフィールド位置交換素子として構成してもよい。

【0058】次に、前記デコード回路16の動作を以下に示す。まず、前記デコード回路16は記憶装置20から復号用データを入力し、IREG6に保持されている命令がオペコードを含んでいるかいないかが分かる。前

記命令がオペコードを含んでいる場合はそのオペコード17により、含んでいない場合は前記復号用データ22により、前記入力端子の交換パターンを決定する。その交換パターンにより各オペランドは次のように出力端子に導かれる。すなわち、前記命令における第1のオペランド18a~dがそれぞれ第1~第4被制御部に対する制御に関連する場合、前記第1のオペランド18aは第1の制御信号に関連する出力端子19aに、第2~第4のオペランド18b~dはそれぞれ第2~第4の制御信号に関連する出力端子19b~dに出力される。また、そのときに使用されなかった4つの端子19b、c、g、hには無印信号を出力させる。命令におけるフィールドビットはそれぞれ等しい幅でなくてもよい。また、フィールドビットがすべて等しい幅の構成の場合は回路が簡単になる利点がある。さらに、信号線はそれぞれ1本で示されているが、複数本の制御線群でも差し支えない。

【0059】次に、図14に本発明におけるデコード回路の第2の実施例の構成を示す。これは、m種の制御群28のうちからn種の制御群26を選択してIMEMビットフィールドに組み込みフィールドアロケーション回路を用いて構成されたものである。n種の入力IN(1)~IN(n)26は、それぞれの信号線と接点を有する出力OUT(1)~OUT(m)28のいずれかの信号線に伝えられ、出力される。図14では、IN(x)、OUT(y)の信号線はそれぞれ1本で示されているが、複数本の制御線群でも差し支えない。図14の構成はPLD等で集積化してもよい。

【0060】図15は、本発明に係る装置をIC化する場合の構成例を示す図である。IMEM4、記憶装置20の内容を本データ処理装置の外から書き替えられる構成となっている。それぞれの枠94、96、98は、半導体集積回路で同一チップに載せる場合を示す。枠94の範囲すなわちIREJ6およびIDEC8fを同一チップに載せて、アプリケーションによってIMEM4のメモリ空間のみ変更してもよい。また、枠96の範囲を同一チップに載せて、アプリケーションによって記憶装置をROMによる書き込みやジャンパ線、ディップスイッチによる固定、半導体集積回路のアルミ配線、PLD等のいずれかの方法を選択してもよい。さらに、枠98の範囲を同一チップに載せて、アプリケーションによって、IMEM4、記憶装置20を命令セットで変更してもよい。この場合、例えば浮動小数点体系の命令セットと固定小数点体系の命令セットのように、互いに排他的な命令セットでのアプリケーションでは、各々、不必要なデコーダフィールドを命令中に持たせなくてもよく、IMEMのビットを有効に利用できる利点がある。なお、本発明は上述した各実施例に限定されるものではなく、その要旨を逸脱しない範囲で、種々変形して実施することができる。

【0061】

【発明の効果】本発明においては、インストラクションデコーダはインストラクションレジスタから入力される命令をデコードする際に、記憶回路から少なくとも1つのデコード用データを取り込み、そのデコード用データによってデコード方法を決定する。前記記憶装置の内容は、デコード用データの変更命令によって書き替えるか、本システムの外部から書き替えるか、あるいは予め固定される。このように構成した場合次のような効果がある。

【0062】まず、前記記憶装置の内容を変えることで、各ビットフィールドとハードウェアの制御線を自由に組み替えることにより、単一命令中に並列制御できる機能ブロック種の組み合わせを自由にできるので、従来よりも大きな汎用性、フレキシビリティ、高速性、さらに設計変更の容易性を与える。

【0063】次に、命令をオペランドのみに構成し、オペコードを前記記憶装置に保持させ、オペコードが変わるときのみオペコード変更命令を実行させるように構成すると従来のオペコードとオペランドとからなる命令を用いた構成に比べて、メモリの容量を小さくでき、コストダウンが可能となり、また、それによってチップサイズが小さくなるので、クロックを早くすることができ、処理の高速化が可能となる。

【0064】加えて、予め複数の命令セットを用意しておき、その命令セットの番号に前記デコード用データのうちの1つのデータを対応させ、そのデータを設定することにより、前記複数の命令セットのうちいずれかの命令セットを用いるかを適宜設定するように構成すれば、アプリケーションにより使用者が命令セットを選択できる高機能なICを提供することができ、メーカーが前記データを出荷前に設定すれば、同一のICを多種のICとして提供することができる。

【図面の簡単な説明】

【図1】本発明の第1の実施例の構成を示すブロック図である。

【図2】2つの型の命令のビット構成を比較した図である。

【図3】図1の装置における命令のビット構成例を示す図である。

【図4】図1の装置におけるインストラクションメモリマップ例を示す図である。

【図5】本発明の第2の実施例の構成を示すブロック図である。

【図6】図5の装置における命令のビット構成例を示す図である。

【図7】図5の装置におけるインストラクションメモリマップ例を示す図である。

【図8】本発明の第3の実施例の構成を示すブロック図である。

【図9】図8の装置における命令のビット構成例を示す図である。

【図10】本発明の第5の実施例の構成を示すブロック図である。

【図11】本発明の第6の実施例の構成を示すブロック図である。

【図12】図10および図11の装置におけるインストラクションメモリマップ例を示す図である。

【図13】本発明におけるデコード回路の第1の実施例の構成を示す図である。

【図14】本発明におけるデコード回路の第2の実施例の構成を示す図である。

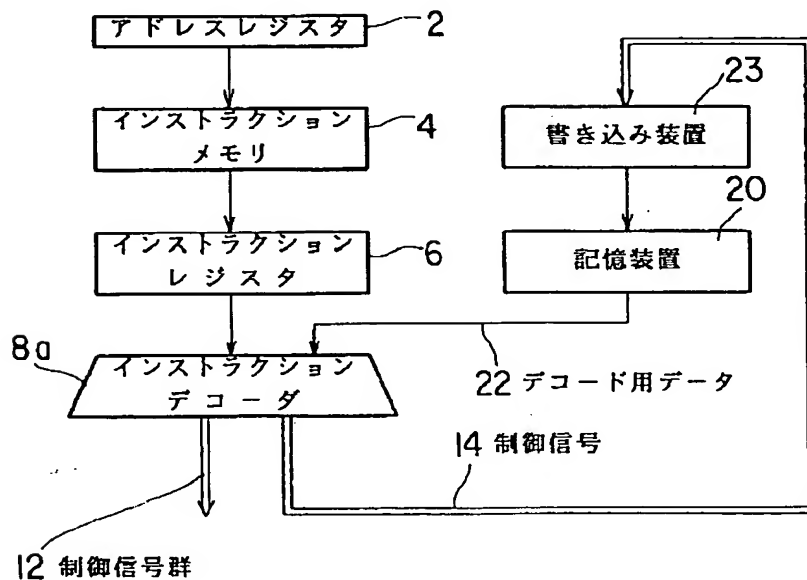
【図15】本発明に係る装置をIC化する場合の構成例を示す図である。

#### 【符号の説明】

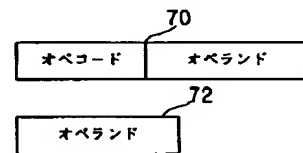
2…アドレスレジスタ、4…インストラクションメモリ、6…インストラクションレジスタ、8a～8f…インストラクションデコーダ、10…被制御部、12…制御信号群、14…制御信号、15…データ書き込み、16…デコード回路、17…命令中のオペコード、18a～18h…命令中の第1のオペランド～オペランド4、19a～19h…各被制御部への第1の制御信号～第8の制御信号、20、21…記憶装置、22…デコード用データ、\*

23…書き込み装置、24a～24c…インストラクションメモリマップ、26…入力IN(1)～IN(n)、28…出力OUT(1)～OUT(m)、30…記憶装置、56a…オペコード変更命令X、56b…オペコード変更命令Y、58a…オペコード変更命令U、58b…オペコード変更命令V、60a…A体系の命令の列、60b…B体系の命令の列、60c…C体系の命令の列、62a…D体系の命令の列、62b…E体系の命令の列、62c…F体系の命令の列、64a…G体系の命令の列、64b…H体系の命令の列、64c…I体系の命令の列、64d…J体系の命令の列、68a、68b…アドレスデータのビット構成、70…第1の型の命令のビット構成、72…第2の型の命令のビット構成、74…A体系の命令のビット構成、76…B体系の命令のビット構成、78…C体系の命令のビット構成、80a…F体系のうちの第1の命令のビット構成、80b…F体系のうちの第2の命令のビット構成、80c…F体系のうちの第3の命令のビット構成、80d…F体系のうちの第4の命令のビット構成、82a…D体系の命令のビット構成、82b…E体系の命令のビット構成、84…第7の命令のビット構成、94…第1の型の同一構成範囲、96…第2の型の同一構成範囲、98…第3の型の同一構成範囲。

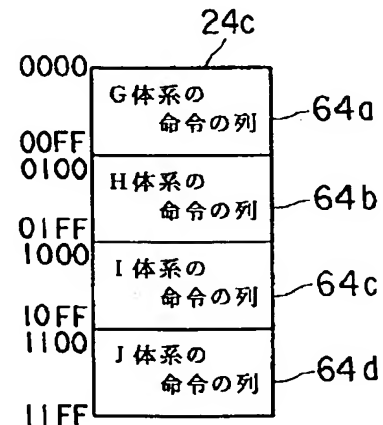
【図1】



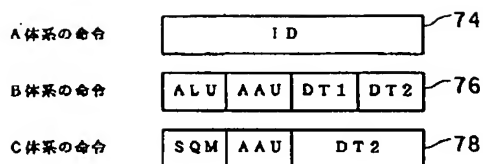
【図2】



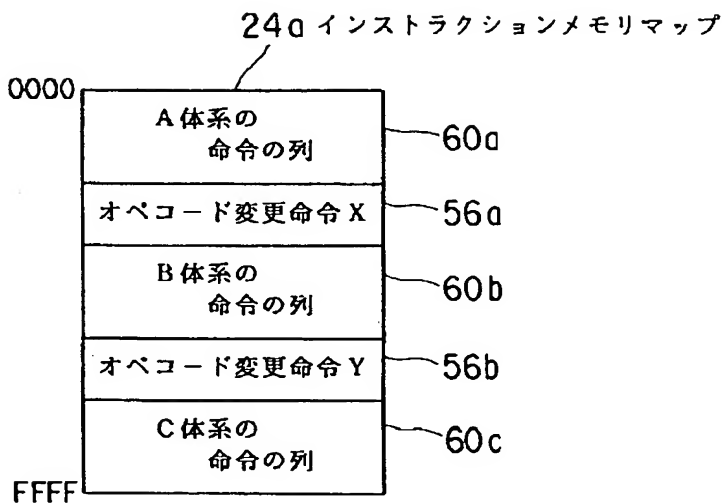
【図12】



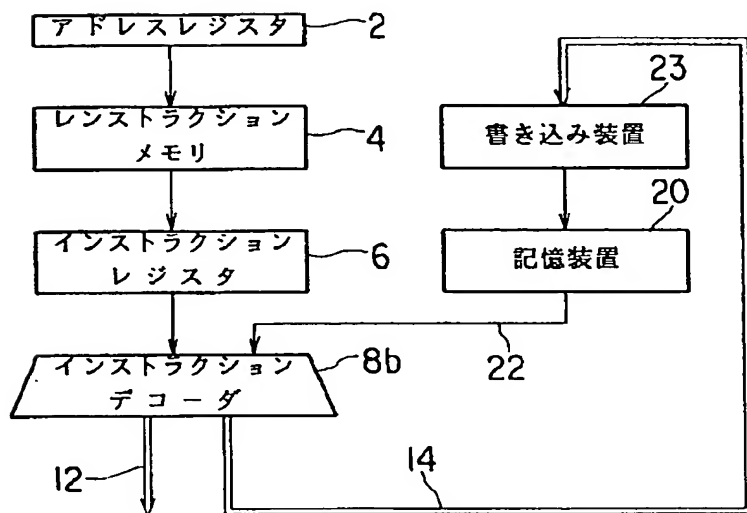
【図3】



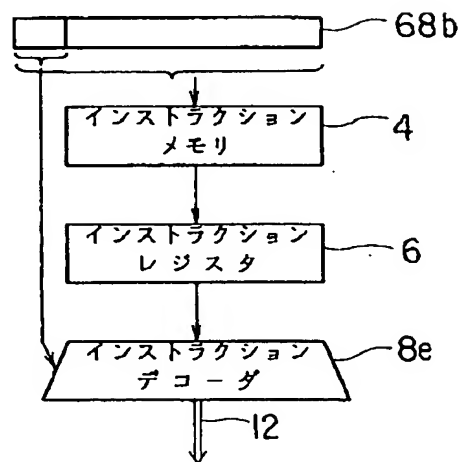
【図4】



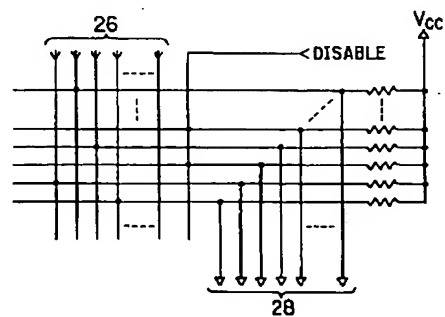
【図5】



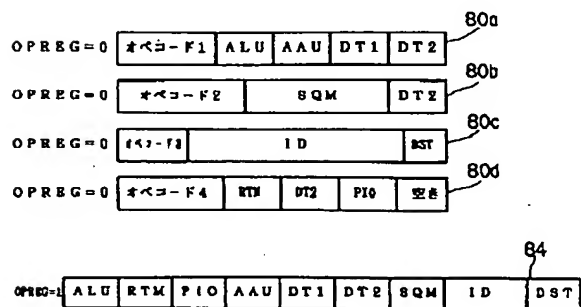
【図11】



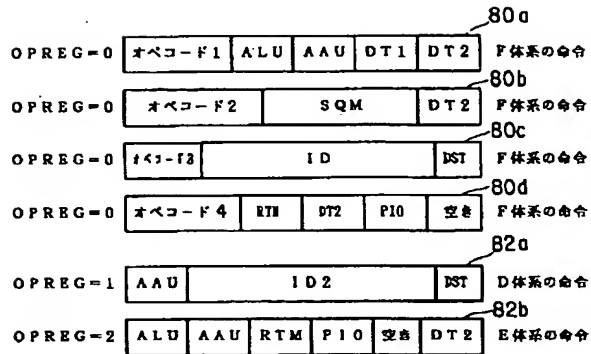
【図14】



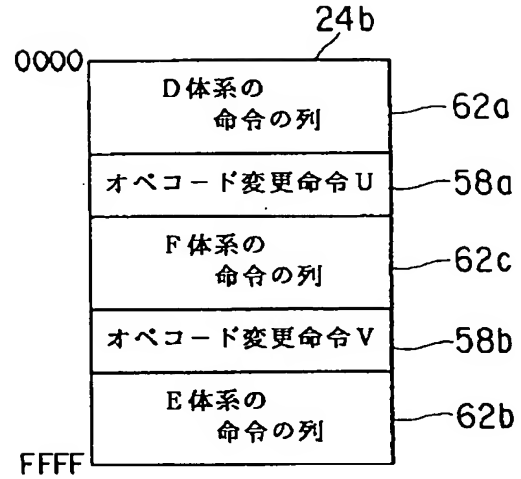
【図9】



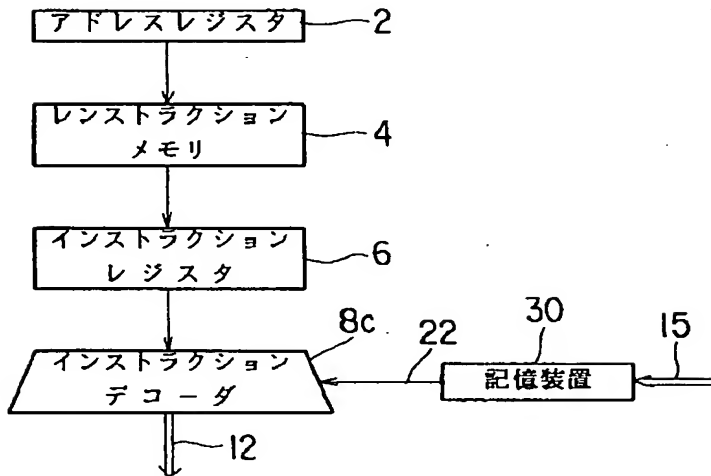
【図6】



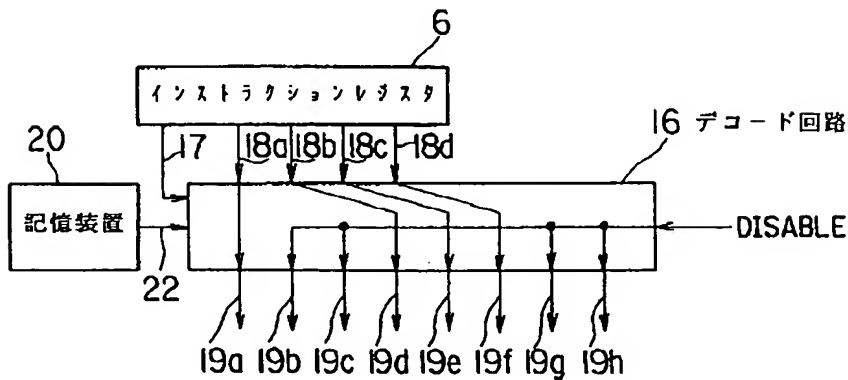
【図7】



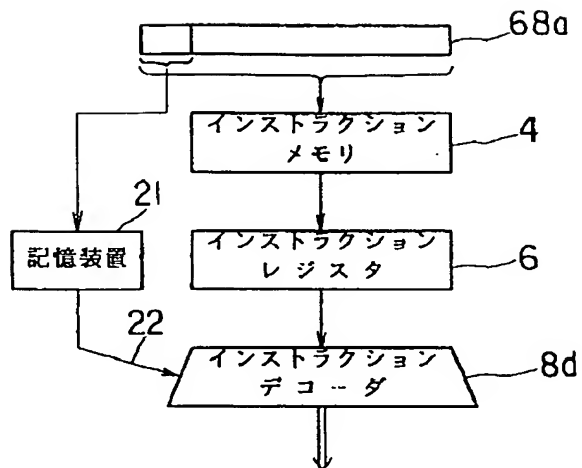
【図8】



【図13】



【図10】



【図15】

